LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

Development of generalized mapping tools to improve implementation of data driven computer simulations (04-ERD-083)

Abelardo Ramirez, Michael Pasyanos, G. Al Franz

September 21, 2004

Disclaimer

Development of generalized mapping tools to improve implementation of data driven computer simulations (04-ERD-083)

Abelardo Ramirez, Michael Pasyanos, G. Al Franz
Lawrence Livermore National Laboratory, Livermore, CA

Introduction:

The Stochastic Engine (SE) is a data driven computer simulation tool for predicting the characteristics of complex systems. The SE integrates accurate simulators with the Monte Carlo Markov Chain (MCMC) approach (a stochastic inverse technique) to identify alternative models that are consistent with available data and ranks these alternatives according to their probabilities. Implementation of the SE is currently cumbersome owing to the need to customize the pre-processing and processing steps that are required for a specific application.

This project widens the applicability of the Stochastic Engine by generalizing some aspects of the method (i.e. model-to-data transformation types, configuration, model representation). We have generalized several of the transformations that are necessary to match the observations to proposed models. These transformations are sufficiently general not to pertain to any single application. This approach provides a framework that increases the efficiency of the SE implementation. The overall goal is to reduce response time and make the approach as "plug-and-play" as possible, and will result in the rapid accumulation of new data types for a host of both earth science and non-earth science problems.

When adapting the SE approach to a specific application, there are various pre-processing and processing steps that are typically needed to run a specific problem. Many of these steps are common to a wide variety of specific applications. Here we list and describe several data transformations that are common to a variety of subsurface inverse problems. A subset of these steps have been developed in a generalized form such that they could be used with little or no modifications in a wide variety of specific applications. This work was funded by the LDRD Program (tracking number 04-ERÐ 083).

Simple data transformations:

The MCMC approach aims to search for subsurface models that are most consistent with available data. These models can then be used to calculate other parameters of interest such as seismic velocity. Some of the calculations require that the properties in a 3D model be somehow "simplified" in order to be consistent with the physics being modeled. Figures 1 - 10 shows some of the simple data transformations we have identified and in many cases have coded.

3D model to 3D volume: (refer to Figure 1). This transformation is simply a reduction in volume size with possible re-sampling. This is necessary when the forward problem requires a full 3-D volume, but not necessarily one that covers the complete 3-D model.

Instead the codes might only require a single parameter (i.e. Vp only instead of Vp, Vs, density, Qp, Qs, etc.) or a small volume along the path, as would be required for finite difference codes. Examples include finite difference codes like E3D (Larsen and Schultz, 1995) or spectral element finite element codes like SPECFEM3D (Komatitsch and Tromp, 1999; Komatitsch et al., 2002) for waveform analysis. This is also required for finite difference travel time codes, like the algorithm originally developed by Vidale (1988) and further refined by Hole and Zelt (1995), which uses an approximation to the Eikonal equation to compute first arrival travel times through regularly gridded velocity structures.

3D model to 2D map: (refer to Figure 2). One such simplification (left side of Figure 2) involves the integration of properties along the vertical direction using depth-dependent sensitivity kernels as weighting functions; the integrated properties are displayed as a 2D map. When the sensitivity kernel has the shape of a delta function with the maximum located at depth Z, this transformation returns the horizontal distribution of model values for this depth. The calculation of seismic surface wave group velocity or phase velocity is an application that uses this approach. This is also used for seismic, body wave travel time calculations to reduce a complex model to a series of 2D surfaces.

3D model to 2D cross-section: (refer to Figure 3). In this case, we need to extract a distribution of values along a 2D vertical plane (say, from source to receiver). In some cases, the 2D distribution of values is calculated by averaging the property along the normal to the plane of interest. Seismic waveform synthesis using a 2-D version of the 3-D finite difference programs like E3D (Larsen and Schultz, 1995) is one application that uses this approach, as would be programs that calculate travel times using a Gaussian beam technique like Xgbm (Davis and Henson, 1993).

3D model to 1D borehole: (refer to Figure 4). This transformation consists of extracting a vertical profile of property values from the 3D model. This approach is used when calculating teleseismic receiver functions, which are the reverberations and conversions of incident P-waves under a seismic station.

3D model to surface: (refer to Figure 5). This data transformation is necessary in order to extract some types of surface from a 3-D model. The surface could either be isocontours like iso-velocities (i.e. surface having 6.0 km/s P-wave velocity) or a surface that might represent physical units such as lithology. Examples of this would be surfaces of basin depth or Moho depth.

2D cross-section to 1D cross-section: (refer to Figure 6). In this case, we extract a distribution of values along a 2D vertical plane, and then average the values to produce a "layered-cake" model. Only the values within a given layer are averaged together in order to preserve the geological layering, layer boundaries and discontinuities. This approach is used when synthesizing waveforms using a simplified 1D model like reflectivity (Kennett, 1985) or a Gaussian beam approach like Xgbm (Davis and Henson, 1993).

3D model to voxel: (refer to Figure 7). This transformation consists of extracting a single voxel value from a 3D model.

2D model to pixel: This transformation consists of extracting a single pixel value from a 2D model.

Integrate properties along a 2-D great circle path: (refer to Figure 8). This transformation integrates a property (such as slowness) along a great circle path. This transformation is useful when calculating seismic arrival times (which accumulates as travel time slowness), group times (which accumulates as group slowness), or amplitudes (which accumulates as attenuation).

Integrate properties along a general 2-D path (refer to Figure 9). This transformation is similar to the one above except that the path of interest follows a general route that is specified by the user. This transformation is useful when large contrasts in seismic properties are expected to result in significant ray bending where the ray is not expected to travel along the great circle path. Examples are the same as in the integration along the great circle path.

Map category to physical data. (refer to Figure 10). We can use the lithology (the general physical characteristics of a rock) as the subsurface model of interest because it is typically correlated with many parameters of interest such as seismic velocity and density. In these cases, one can choose a categorical simulation approach where each category is associated with a discrete resistivity value. For example, if the set of categories is $\{gravel, sand, clay\}$, then $gravel \rightarrow VP_1, sand \rightarrow VP_2, clay \rightarrow VP_3$, where the VP_i represent distinct p-wave velocity values or some other property. This mapping from category to physical parameter is one of the data transformations we could generalize in order to accelerate the implementation of specific applications of the MCMC approach.

Advanced data transformations:
Transform from one physical property to another when multiple data sets are inverted jointly. The MCMC method simultaneously uses many types of data to refine our understanding of complex subsurface systems. Suppose that the system under study consists of a plume of fluid injected into a subsurface layer. The actual connection of a plume model to an observation is made via a forward model: given a possible plume configuration, the forward model predicts the values that would be observed by actual measurement. Then, the predicted data are compared to the measured data, yielding an estimate of the probability that the proposed plume model is in fact the true model. By staging these comparisons in a series, we can identify probable plume models that are consistent with all available data. For example, consider an application where CO_2 is injected into a geologic reservoir and that cross-borehole electrical resistance and borehole tiltmeter surveys have been made to monitor the spatial extent of the CO_2 flood. To solve the forward problem for the electrical survey data, it is necessary to transform a given plume configuration into a 3D model of electrical resistivities using an appropriate

petrophysical model. To solve the forward problem for the tiltmeter data, we need to transform the plume model to a 3D pressure field using a different petrophysical model. In a staged inversion, these data transformations could be generalized such that a single software module can be used regardless of the type of data available and the petrophysical model needed.

Pre-processing of data differences. In some applications, the goal of the inversion is to characterize differences that develop due to some forcing condition such as injecting fluid into a layer, or due to different frequencies being used to probe the target. In these situations, the observations of interest are not the measured quantities themselves, but rather, time (or frequency) differences in the data. This type of perturbation analysis belongs to the class of time-dependent, 4D problems (function of x, y, z, and t) rather than the simpler 3D, time-independent problems. An example of a 4D problem would be the use of INSAR (interferometric synthetic aperture radar) data used to monitor small changes in ground surface elevation due to tectonic or man-made events. We could consider the development a general module that any application could use to generate the data differences.

Modeling of realistic spatial distributions of a physical parameter:
Natural geologic systems exhibit spatial variability of physical properties such as seismic velocity. This spatial variability is typically not random, i.e., it exhibits trends that are related to the various geologic processes that produced or distributed the geologic materials. When these trends can be quantified, they can be used as a constraint to stabilize the MCMC inversion. For example, in sedimentary basins, layers of gravel, sand, silt and clay tend to be deposited in a particular order, and tend to have particular correlation lengths. In many geologic settings, available data suggests that the values of a given property vary in a gradual or "smooth" way from one location to the next. We can make use of these trends to produce subsurface models that honor these observations. The models produced can also honor data collected along a profile, and spatial correlation data.

Honor vertical profile information: When data collected along a profile are available (e.g., borehole geophysical logs, core sample properties), we can use it to constrain the properties of the subsurface models proposed by the MCMC process. The goal of this approach is to produce models that exhibit the same properties along the line where the profile data was observed. The profile data can be considered as 100% accurate, and thus the proposed model has a probability $p = 1.0$ that it will honor the data exactly. The data can also be considered as somewhat uncertain and the models proposed have a probability $0.0 < p \leq 1.0$ that it will honor the data within some tolerance; in this case p is inversely related to the uncertainty.

Honor surface information (maps): This approach is similar to the one above; in this case, the goal is to produce models that exhibit the properties observed along the plane located at the ground surface. As before, the data can be considered as 100% accurate or have some measure of uncertainty.

Honor smooth trend - spatial correlations in a local neighborhood: (refer to Figure 11). Many subsurface environments tend to exhibit properties that vary smoothly between a given locality and its adjacent neighbors. When inverting for such models it is desirable to include a constraint that forces the search to only consider smoothly varying models. In these situations, one can introduce a dependency between the local value (e.g., voxel or profile) and that of its neighbors. When the pixels consist of continuous values, the simplest approach is the neighborhood averaging. Each pixel is replaced by the average value of the pixels contained in some neighborhood about it. This approach tends to blur the image. An alternative approach is to use a median filter. In this case, a neighborhood around the pixel under consideration is used, but this time the pixel value is replaced by the median pixel value in the neighborhood. This approach has the advantage that the sharpness of image edges is preserved.

In cases where the image values consist of categories (e.g., sand, gravel and clay) we can consider the following approach. If a given voxel consists of sand, there is a probability p such that $0.0 < p \leq 1.0$ that the surrounding voxels also consist of sand. We can impose this type of smoothness constraint by forcing adjacent regions to match identically (or within some tolerance) a given voxel or profile. The values of p can be selected using prior data (data that can be used to compute or to sample the prior probability distribution) and expert judgment. We could develop a generalized module that incorporates one or more of these approaches.

Honor spatial correlation data – Fourier transform approach: This method can also produce realistic models of subsurface heterogeneity using prior knowledge. It is conceptually similar to the previous approach but is implemented by characterizing correlation length scales in the frequency domain. This method models variability as a stochastic process that can be characterized by its spatial covariance structure (e.g. Hubbard et al., 1999). Spatial covariance is used to determine the degree to which two or more spatial random variables are related (e.g. the thickness of sand and clay layers) The spatial covariance structure is based on data such as core sample observations, outcrop maps and the distances separating the observations. It assumes that the subsurface model is composed of variations characterized by different length scales. The spatial covariance function is expressed as a spectral density function using Fourier transformation.

Honor spatial correlation data and profile information – TSIM: TSIM is a categorical geostatistical approach that can produce realistic models of subsurface heterogeneity using prior knowledge (Carle, 2003). This methodology combines disparate types of observational data such as geophysical borehole logs, geologic insight, and geostatistical trends to produce a consolidated body of knowledge indicating those layer configurations that are most consistent with the available data. It makes use of data such as the number and type of materials (e.g. gravel, sand, clay), the relative volumetric proportions for each of the materials, and horizontal and vertical correlation lengths. It uses the transition probability statistic to analyze spatial variability of geologic materials and to formulate co-kriging equations. This approach makes use of spatial cross correlations (how

different geologic materials tend to locate in space relative to each other). It also honors profile information in a way that is more realistic than described previously under "Honor vertical profile information". TSIM assumes that the profile data has a measure of uncertainty; thus, the probability that a given model will honor the profile data is no longer 1.0; instead it is inversely proportional to the data uncertainty.

Post-processing model transformations:
As the Markov chain generates samples, it is important to verify that these samples are statistically representative of the posterior distribution. The methodology used to perform this verification is referred to as "convergence analysis". Convergence analysis tools developed during a previous project assumed that the posterior samples consisted of 2D or a 3D distribution of values within a mesh; all the mesh elements are assumed to be of the same size.

The same assumption is made by the tools previously developed for "posterior analysis"; i.e., the tools used to estimate the properties of samples of the posterior distribution produced by the MCMC approach. These tools summarize the relevant information in the posterior samples so that it can be visualized and understood.

Transform a columnar model to a 3D mesh model: (refer to Figure 12). An application of interest produces models where the properties of the domain are defined by a series of layers arranged within columns. Multiple columns specify the distribution of seismic property values in 3D space. To use existing tools, it is necessary to transform from one parameter space (column model) to another (3D mesh model). We are developing the tools necessary to perform this transformation.

Probability distribution transformations:
There are several probability distribution used when generating various types of random variables used by the MCMC process. In general, these require transforming from one type of distribution to another such as transforming a uniform distribution to a normal (or vice versa), or a general distribution to a standard distribution (or vice versa). We have used the following transformations for our work:

    a) *Transform a general normal distribution to a standard normal distribution.*
    b) *Transform a standard normal distribution to a standard uniform distribution.*
    c) *Transform a standard uniform distribution to a standard normal distribution.*
    d) *Transform a standard normal distribution to a general normal distribution.*

Implemented transformations:
Several codes have been constructed to implement some of the transformations listed above. Most of these programs have been coded in the popular Python programming language. Python is an object-oriented language used for both stand-alone programs and scripting applications (Lutz and Ascher, 1999). The few that are written in "C" are identified by a "*".

Transformation:	Module name:	Class/method:

3D model to 3D volume		
3D model to 2D map	starting_model_for_swaves.py forward_body_script.py	Surface_Waves Project
3D model to 2D cross-section	layers08.py	Layer_Plot.ConvertLayersToXYZ
3D model to 1D borehole	models.py	Model.get_vertical_profile
3D model to surface	forward_body_script.py	ForwardBody.Project
2D cross-section to 1D cross-section		
3D model to voxel		
2D grid to pixel	starting_model_for_swaves.py	Starting_Model.Surface_Waves
Integrate properties along a 2-D great circle path	BodyForward.c SurfForward.c	BodyForward SurfForward
Integrate properties along a general 2D path		
Integrate properties along a general 3D path	forward_body_script.py	Forward_Rcvr.Conversion
Map category to physical data		
Transform from one physical property to another when multiple data sets are inverted jointly		
Pre-processing of data differences		
Honor vertical profile information	forward_rcvr_script.py	Forward_Rcvr.Conversion
Honor surface information (maps)		
Honor smooth trend - spatial correlations in a local neighborhood	models.py	Model.NeighborhoodWeightedAverageFilter
Honor spatial correlation data – Fourier transform approach		
Honor spatial correlation data		

and profile information – TSIM		
Transform a columnar model to a 3D mesh model	super_script_meshmodel_v2_file_list.py	chng_col2grid
Transform general normal distribution to a standard normal distribution	Sampler.c*	RDist()
Transform standard normal distribution to a standard uniform distribution	Sampler.c*	P()
Transform standard uniform distribution to a standard normal distribution	Sampler.c*	UniformToNormal()
Transform a standard normal distribution to a general normal distribution	Sampler.c*	RDist()

Table 1 lists the transformations that have been implemented as part of this project.

<u>Applications:</u>
Here we present how we are currently employing some of these transformations in an application to construct regional models of the crust and upper mantle for the Yellow Sea – Korean Peninsula (YSKP) region, using three different data types (receiver functions, surface wave dispersion, body wave travel times). A detailed description of this work has been submitted for publication in Geophysical Research Letters: "Reconciling data using Markov Chain Monte Carlo: An application to the Yellow Sea - Korean Peninsula region" (Pasyanos et al., 2004). Below we list the various steps in the processing sequence and the transformations used for the YSKP work.

Base Sampler
        Honor smooth trend

Stage 1 (Receiver functions)
        3D model to 1D borehole

Stage 2 (Surface wave dispersion)
        3D model to 2D map
        Integrate along great circle path

Stage 3 (Body wave travel times)
      3D model to 2D map
      3D model to surface
      Integrate along great circle path

Convergence analysis
      Transform a columnar model to a 3D mesh model

Clustering analysis
      Transform a columnar model to a 3D mesh model

While we have primarily focused our efforts on the YSKP application, we can now easily apply the transformations developed to other geophysical problems. We now present an example of the procedure for the construction of a seismic model of the Las Vegas Valley and southern Nevada using data sets available for this region. This model is of interest for predicting ground motions for the Test Site Readiness Program. The order of the stages listed is arbitrary.

Base Sampler
      Honor smooth trend

Stage 1 (Geotechnical shear wave profiles)
      3D model to 1D borehole

Stage 2 (Seismic reflection profiles)
      3D model to 2D map
      3D model to surface
      Integrate along great circle path

Stage 3 (Receiver functions)
      3D model to 1D borehole

Stage 4 (Surface wave dispersion)
      3D model to 2D map
      Integrate along great circle path

Stage 5 (Body wave travel times)
      3D model to 2D map
      3D model to surface
      Integrate along great circle path

Stage 6 (Gravity)
      3D model to 2D map

Convergence analysis

Transform a columnar model to a 3D mesh model

Clustering analysis
Transform a columnar model to a 3D mesh model
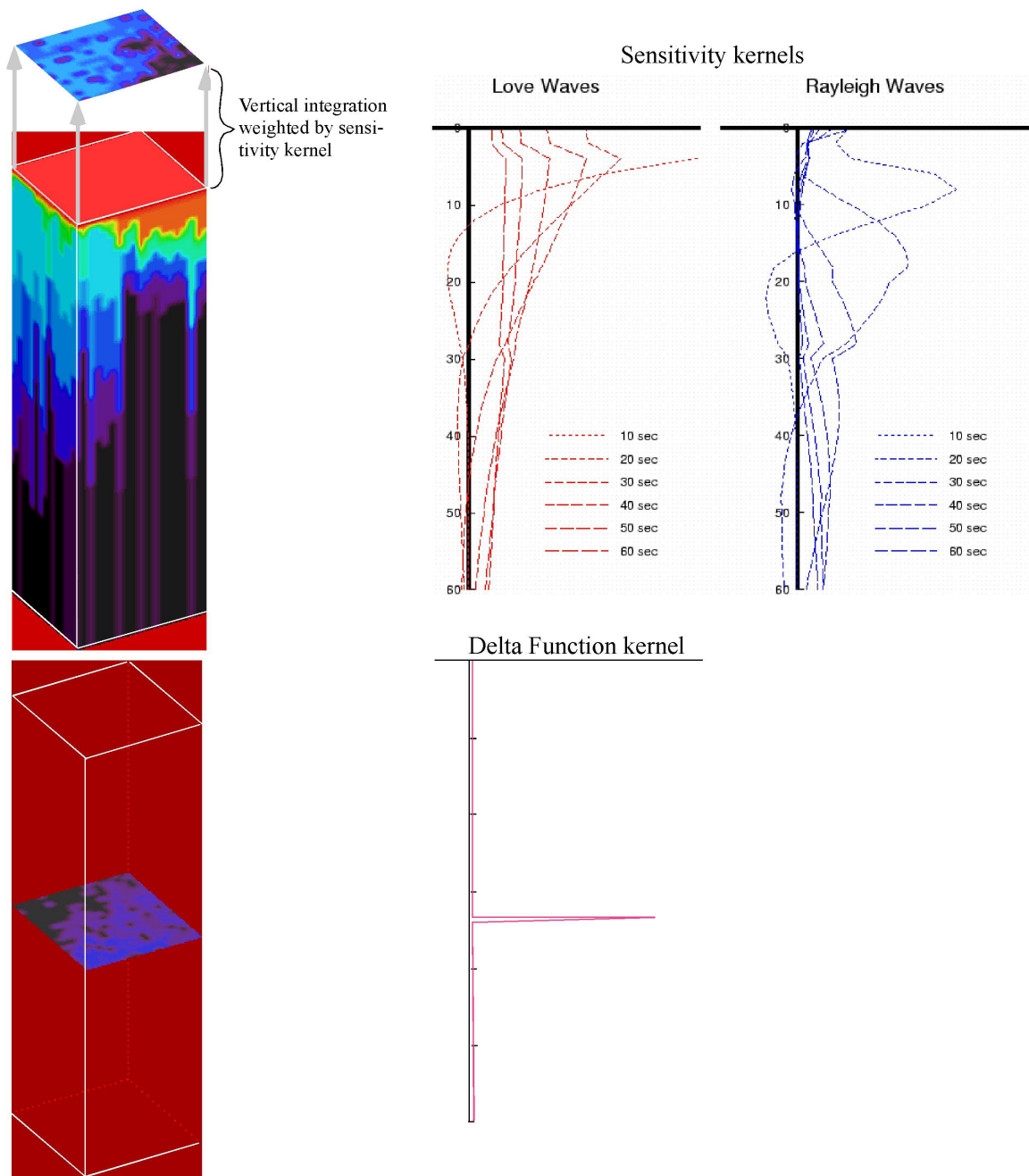
Figures:
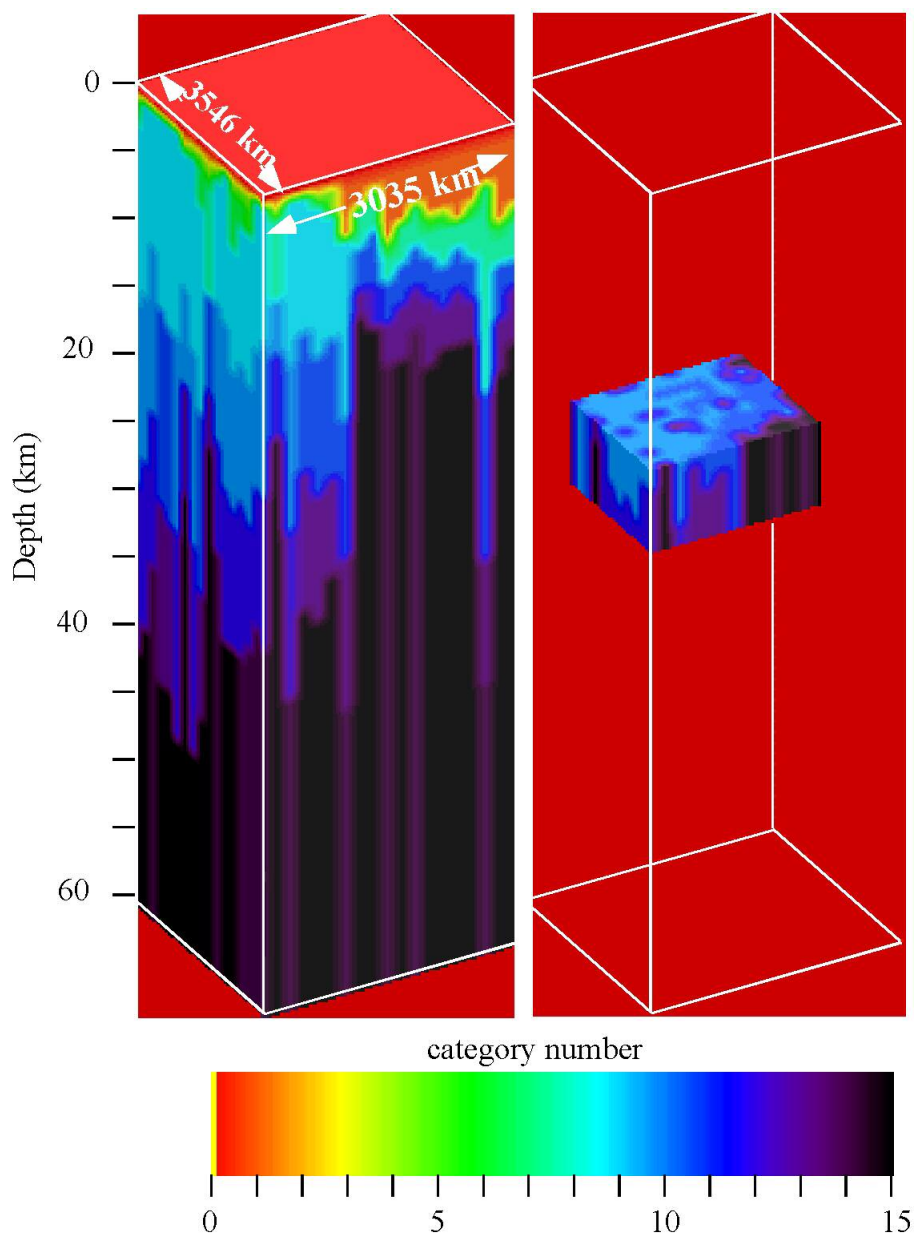
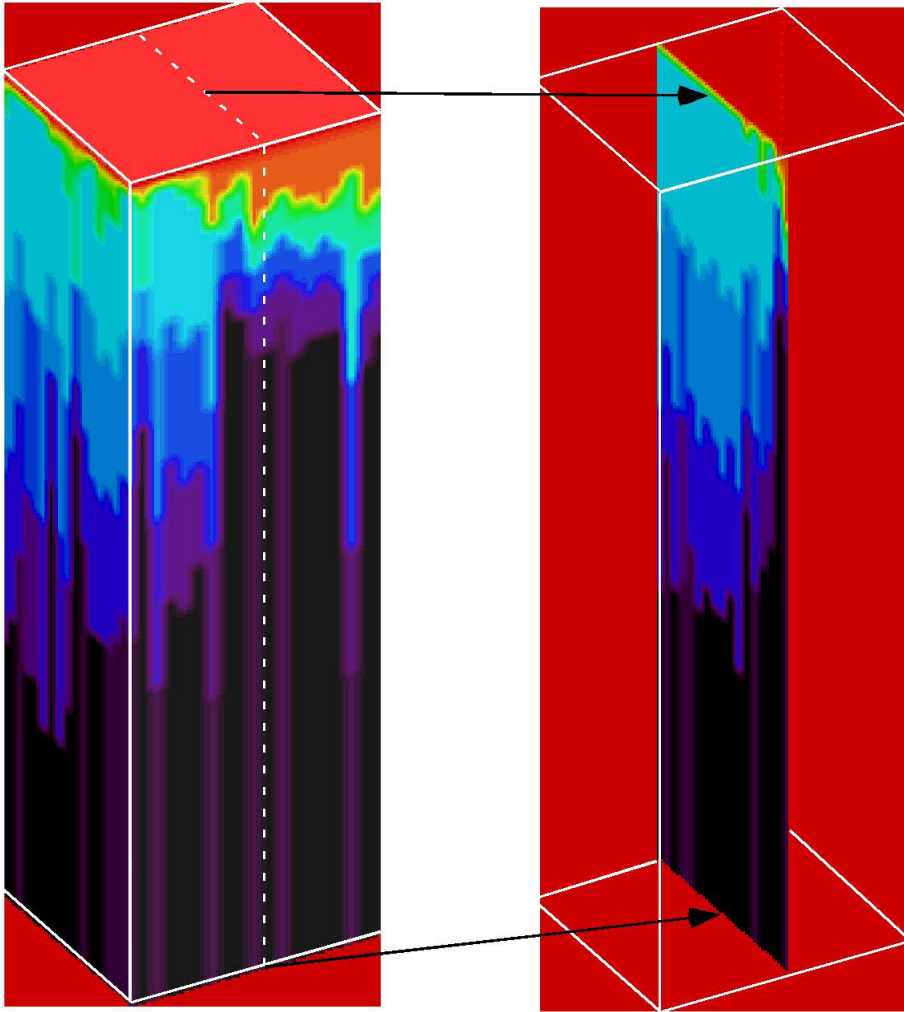Figure 1: 3D model to 3D volume

Figure 2: 3D model to 2D map

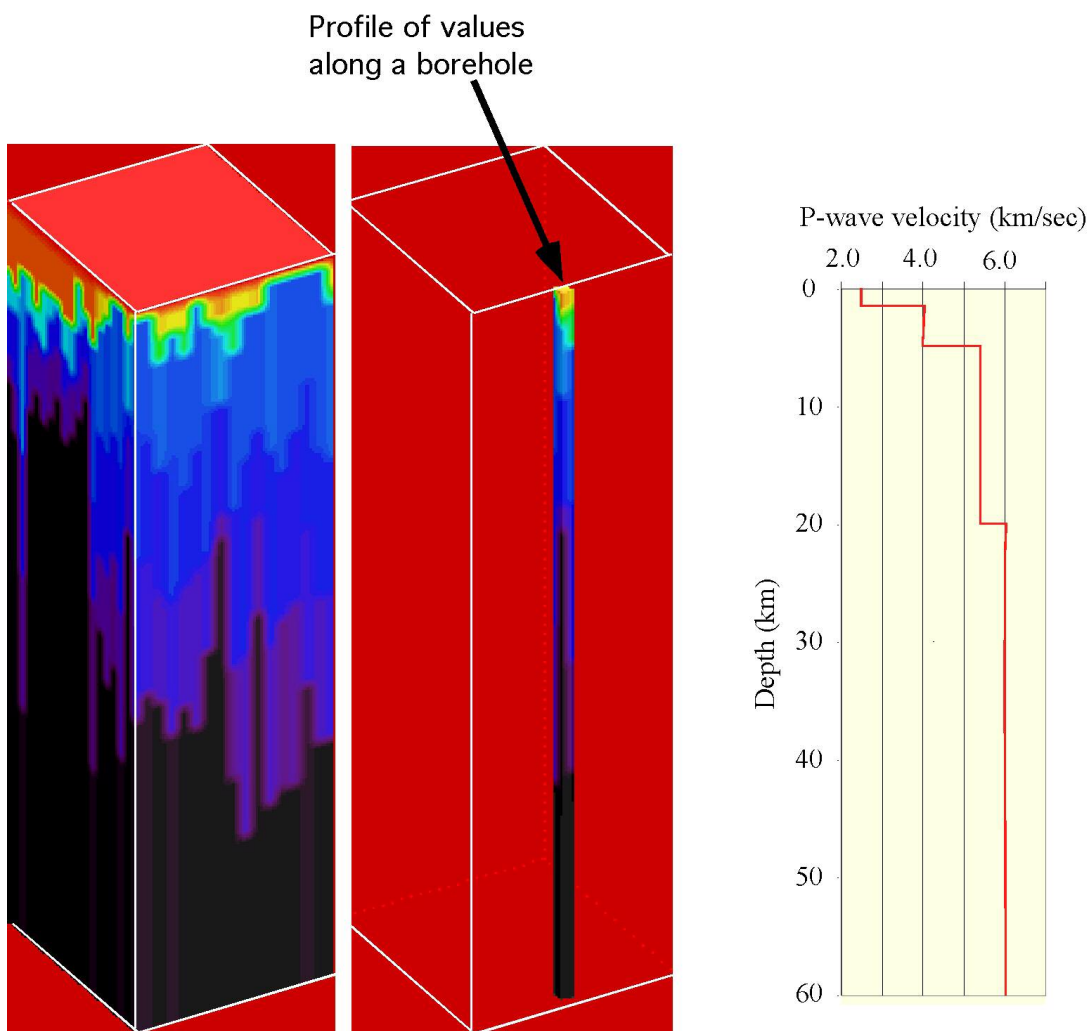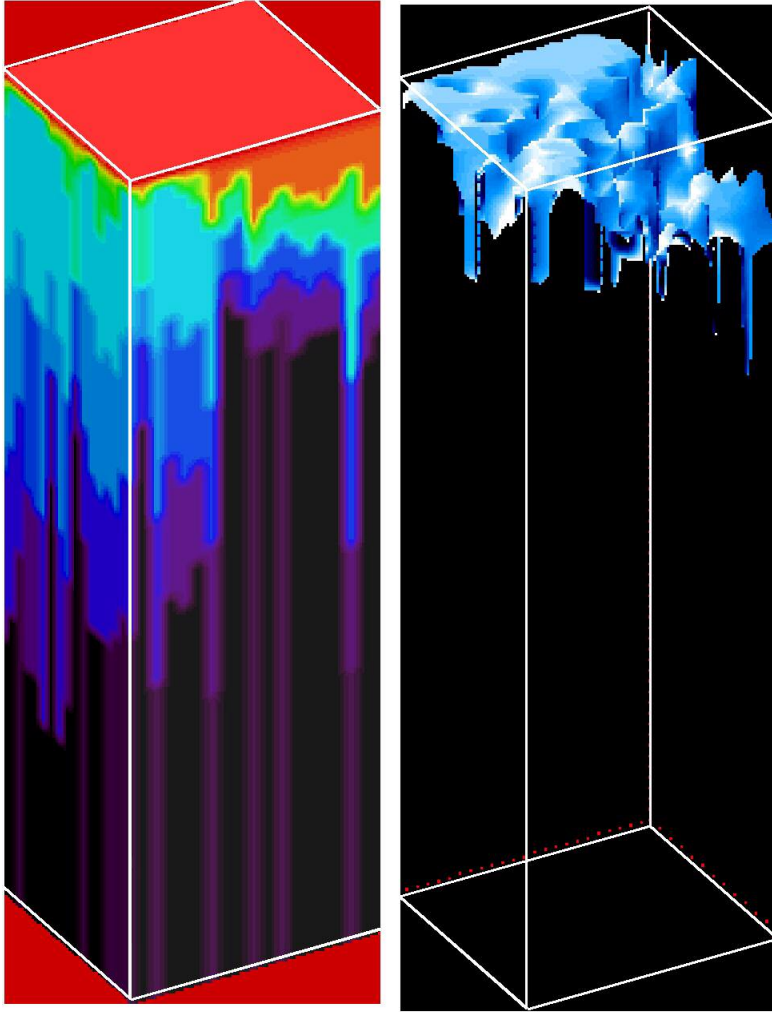Figure 3: 3D model to 2D cross-section

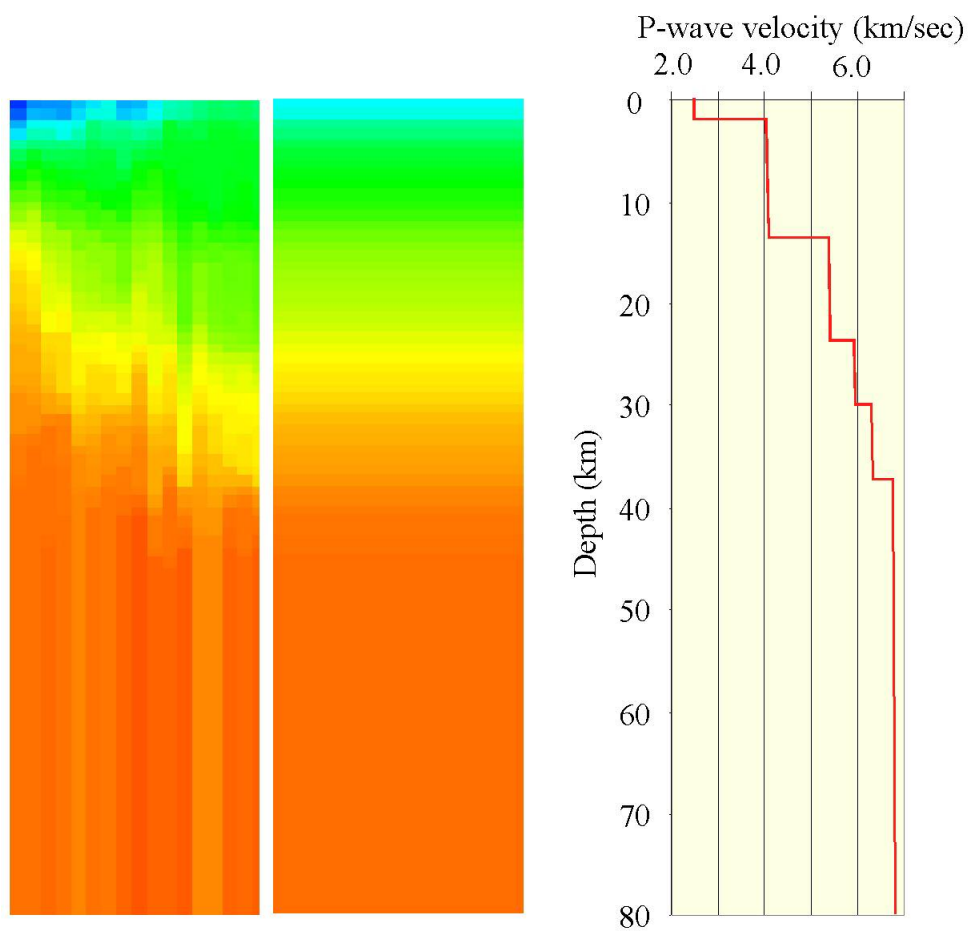Figure 4: 3D model to 1D borehole

Figure 5: 3D model to surface

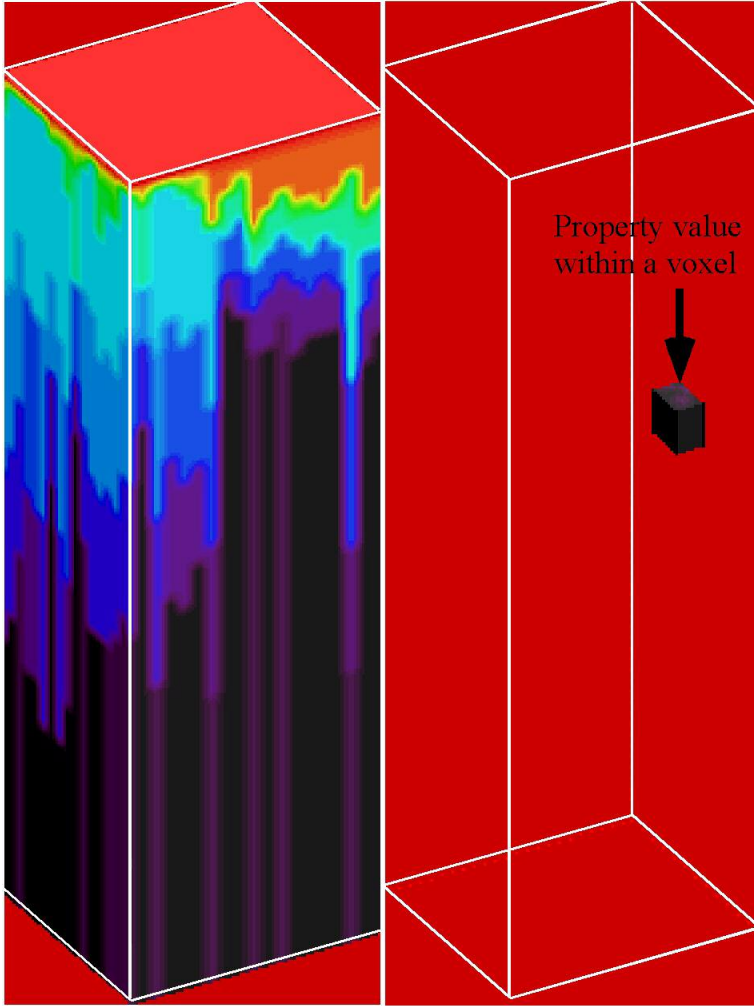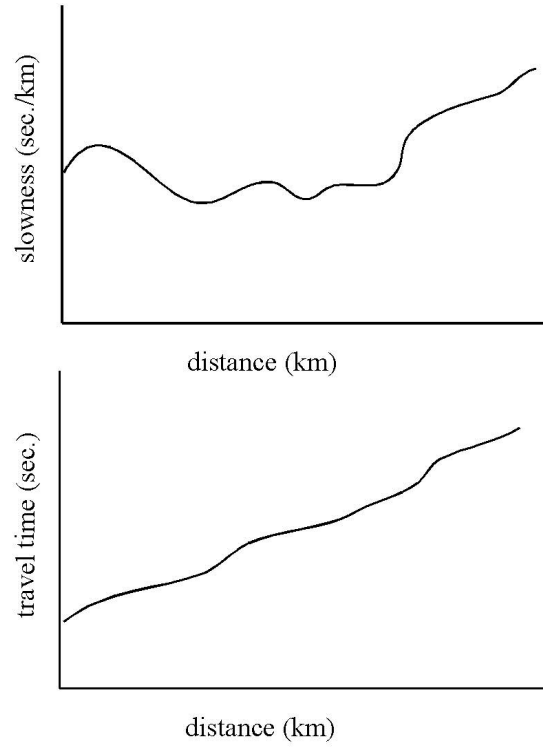Figure 6: 2D cross-section to 1D cross-section

Property value
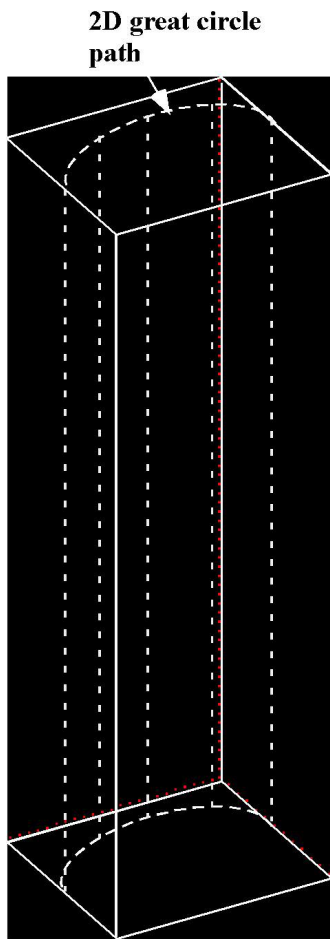within a voxel

Figure 7: 3D model to voxel

Figure 8: Integrate properties along a 2D great circle path

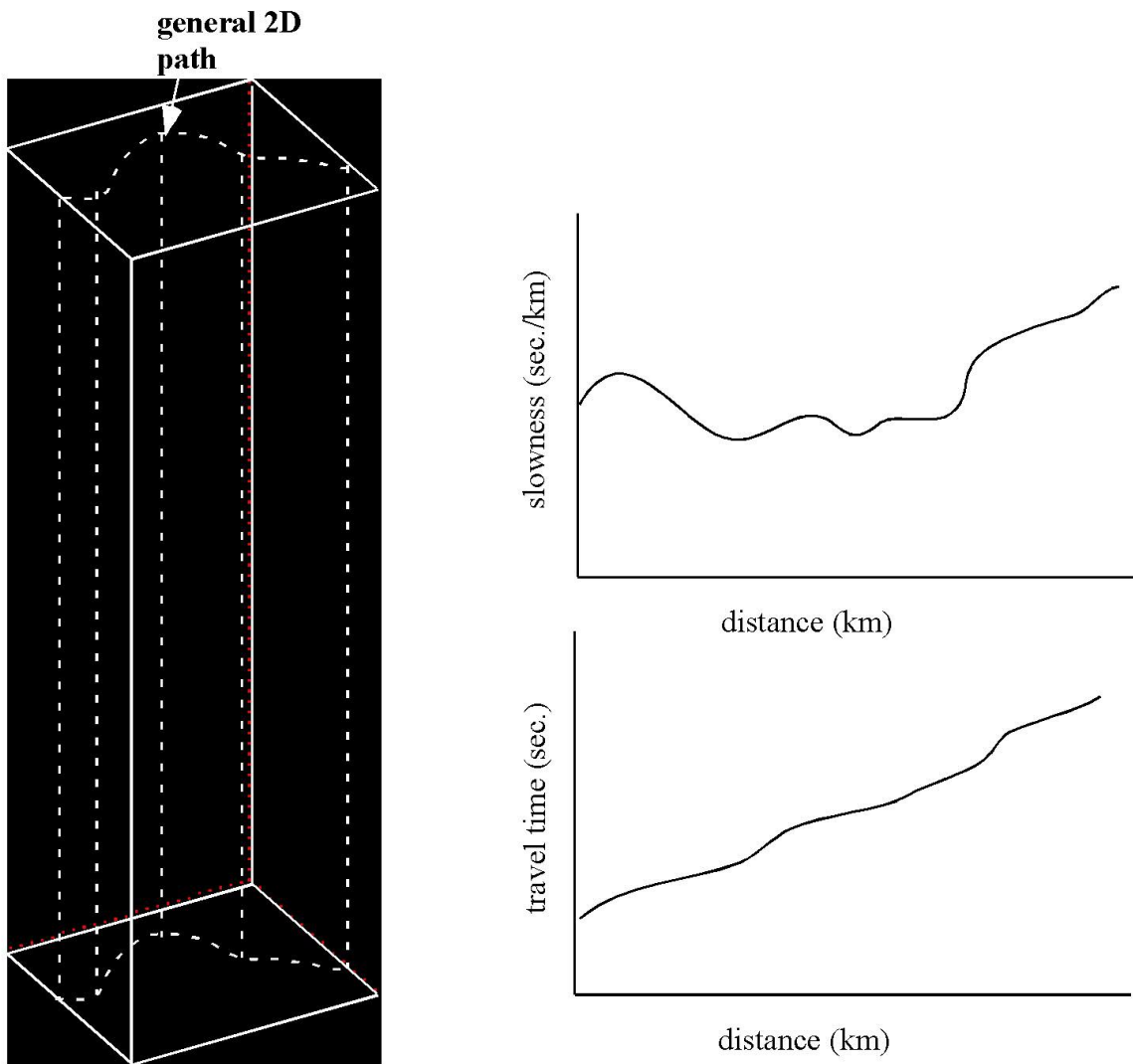Figure 9: Integrate properties along a general 2D or 3D path

Figure 10: Map category to physical data values
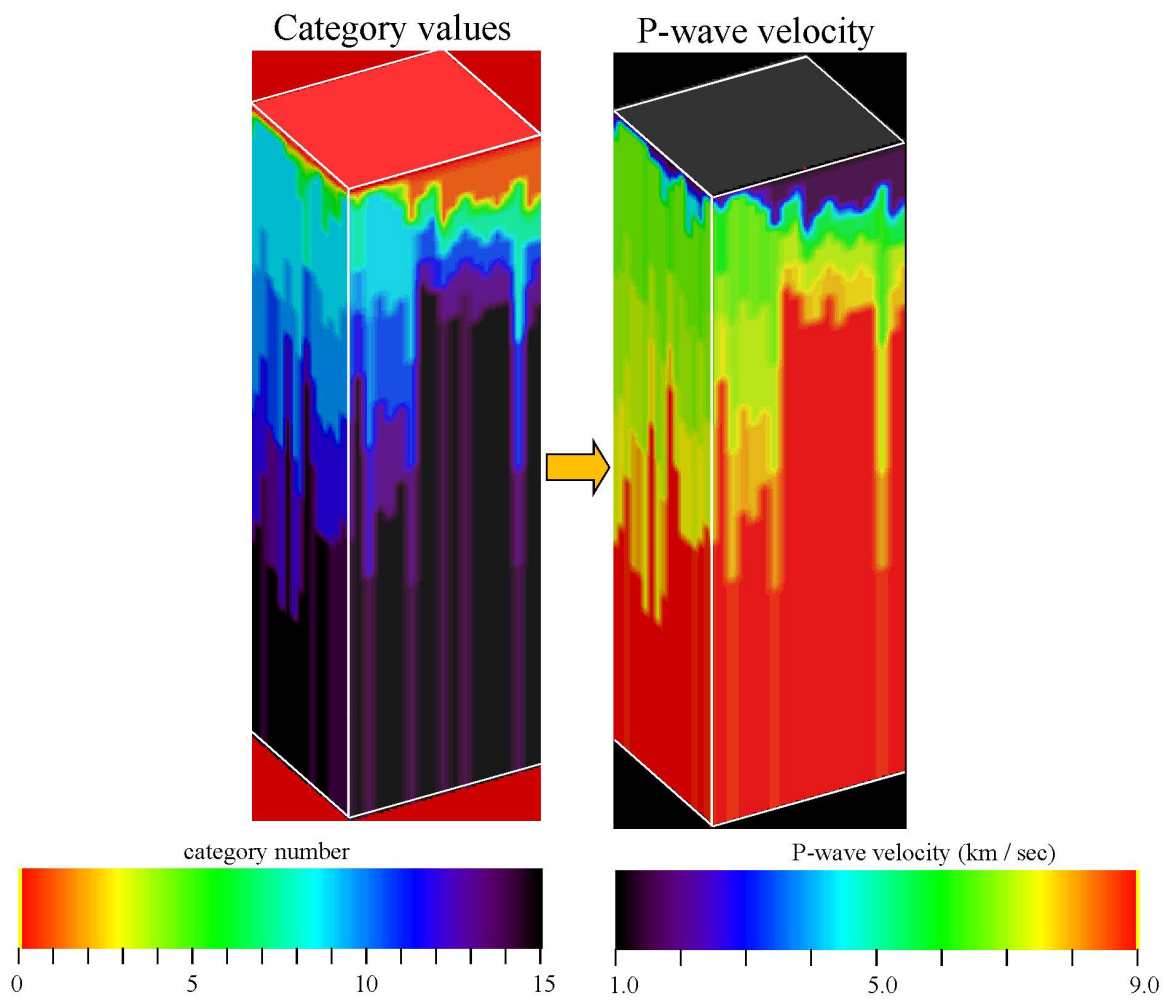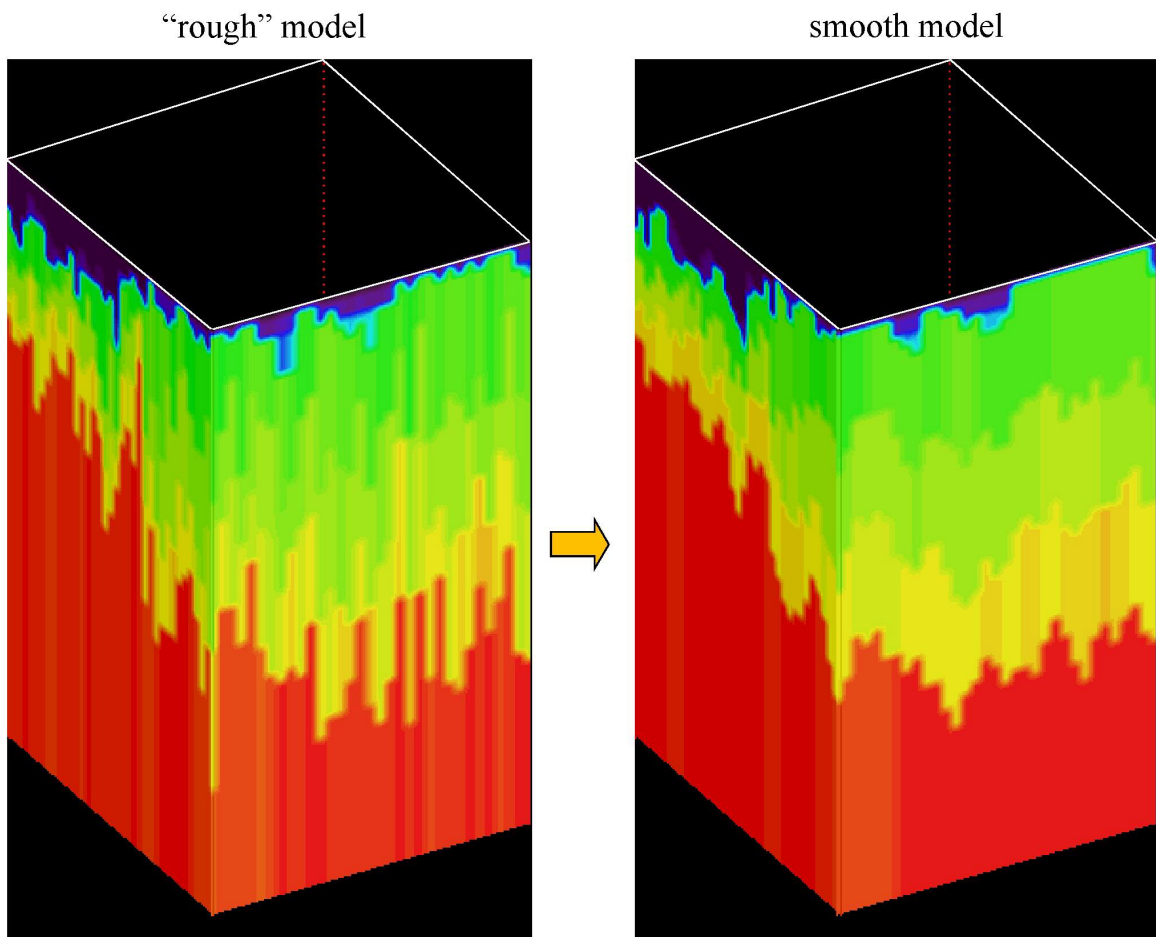
"rough" model   smooth model

Figure 11: Transform a "rough" model to a model whose
properties vary smoothly from one location to the next

columnar model

3D mesh model

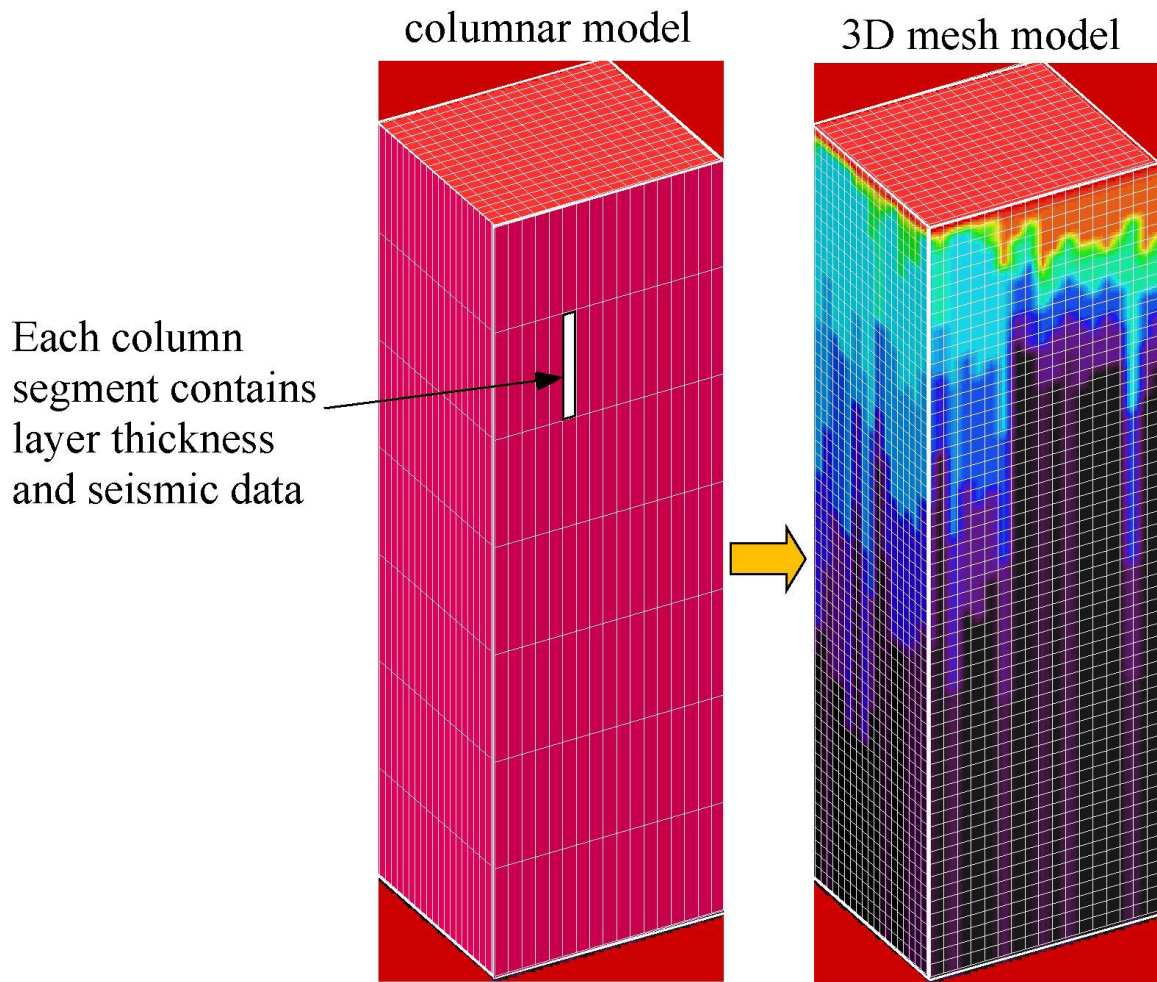Each column
segment contains
layer thickness
and seismic data

Figure 12: Transform a columnar model to
a 3D mesh model

 References:

Carle, S. (2003), Integration of soft data into categorical geostatistical simulation, UCRL-JC-152653, Lawrence Livermore National Laboratory, Livermore, CA.

Crotwell, H.P., T.J. Owens, and J. Ritsema (1999), The TauP ToolKit: Flexible seismic travel-time and raypath utilities, *Seism. Res. Lett.,* 70, 154-160.

Davis. J.P. and I.H. Henson (1993), User's Guide to Xgbm: An X-Windows System to compute Gaussian beam synthetic seismograms. Teledyne Geotech Alexandria Laboratories, Alexandria, VA, 1.1 edition.

Hole, J.A. and B.C. Zelt (1995), 3-D Finite difference reflection travel times, *Geophys. J. Int.*, 121, 427-434.

Hubbard, S., Y. Rubin, and E. Majer, 1999, Spatial correlation structure estimation using geophysical and hydrogeological data, *Water Resources Research*, vol. 35, pp. 1809-1825.

Kennett, B.L.N. (1985), *Seismic wave propagation in stratified media*, 342 pp., Cambridge University Press, Cambridge, England.

Komatitsch, D. and J. Tromp (1999). Introduction to the spectral element method for three-dimensional seismic wave propagation, *Geophys. J. Int.,* 139, 806-822.

Komatitsch, D., J. Ritsema and J. Tromp (2002). The Spectral-Element Method, Beowolf computing and global seismology, *Science*, 298, 1737-1742.

Larsen, S.C. and C.A. Schultz (1995), ELAS3D: 2D/3D Elastic Finite-Difference wave propagation code, Lawrence Livermore National Laboratory, 19pp.

Lutz, M. and D. Ascher, 1999, *Learning Python*, O'Reilly and Assoc., Sebastopol, 366pp.

Pasyanos, M.E. , G.A. Franz, and A.L. Ramirez (2004), Reconciling data using Markov Chain Monte Carlo: An application to the Yellow Sea - Korean Peninsula region, submitted to *Geophys. Res. Lett*.

Ramirez, A. L., J.J. Nitao, W.G. Hanley, R.D. Aines, R.E. Glaser, S.K. Sengupta, K.M. Dyer, T.L. Hickling, W.D. Daily, 2004, Stochastic Inversion of Electrical Resistivity Changes Using a Markov Chain, Monte Carlo Approach, submitted to *Journal of Geophysical Research* , UCRL-JRNL-155048 Rev. 2, Lawrence Livermore National Lab., Livermore CA.

Vidale, J., (1988) Finite-difference Calculation of Travel Times, *Bull. Seismol. Soc. Am.*, 78, 2062-2076.